

# Modulhandbuch

---

## Master

# Research in Computer & Systems Engineering

---

**Studienordnungsversion: 2016**

**gültig für das Sommersemester 2019**

Erstellt am: 02. Mai 2019

aus der POS Datenbank der TU Ilmenau

Herausgeber: Der Rektor der Technischen Universität Ilmenau

URN: urn:nbn:de:gbv:ilm1-mhb-14547

# Inhaltsverzeichnis

[illegible]

## Algorithms

Fachabschluss: Prüfungsleistung mündlich 30 min

Art der Notengebung: Gestufte Noten

Sprache:english

Pflichtkennz.:Pflichtfach

Turnus:Sommersemester

Fachnummer: 101720

Prüfungsnummer:2200604

Fachverantwortlich: Prof. Dr. Martin Dietzfelbinger

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):105			SWS:4.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2242																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
	2	1	1																											

### Lernergebnisse / Kompetenzen

**Fachkompetenz:** The students know the basic principles of the design and the analysis of algorithms: correctness and running time. They know the  $o$  notation and their use for analyzing running times. They know basic number theoretical algorithms (addition, multiplication, division, modular multiplication, modular exponentiation, greatest common divisor), they know basic primality tests and the RSA scheme. The students know the divide-and-conquer paradigm with the master theorem (and its proof) and the most important examples like Karatsuba's algorithm, Strassen's algorithm, Mergesort, Quicksort, and the Fast Fourier Transform. They know basic techniques for orienting oneself in graphs and digraphs: BFS, DFS, Kosaraju's algorithm for strongly connected components. They know Dijkstra's algorithm for calculating shortest paths in graphs, and the data type priority queue with its most important implementation techniques "binary heap" and "d-ary heap". Out of the family of greedy algorithms they know Kruskal's algorithm and Prim's algorithm for the problem of a minimum spanning tree, including the correctness proof and the runtime analysis including the use of the union find data structure. As another greedy algorithm they know Huffman's algorithm for an optimal binary code. In the context of the dynamic programming paradigm the students know the principal approach as well as the specific algorithms for Edit distance, all-pairs shortest paths (Floyd-Warshall), single-source shortest paths with edge lengths (Bellman-Ford), knapsack problems and matrix chain multiplication. They know the basic definitions and facts from NP-completeness theory, in particular the implications one gets (if  $P \neq NP$ ) from the fact that a search problem is NP-complete as well as central examples of NP-complete problems.

**Methodenkompetenz:** The students can formulate the relevant problems and can describe the algorithms that solve the problems. They are able to carry out the algorithms for example inputs, to prove correctness and analyze the running time. They are able to apply algorithm paradigms to create algorithms in situations similar to those treated in the course. They can explain the significance of the concept of NP-completeness and identify some selected NP-complete problems.

### Vorkenntnisse

Basic Data Structures, Calculus, Discrete Structures

### Inhalt

Fibonacci numbers and their algorithms, Big-O notation, multiplication, division, modular addition and multiplication, fast exponentiation, (extended) Euclidean algorithm, primality testing by Fermat's test (with proof) and by Miller-Rabin (without proof), generating primes, cryptography and the RSA system (with correctness proof and runtime analysis). The divide-and-conquer scheme, Karatsuba multiplication, the master theorem (with proof), Mergesort, Quicksort, polynomial multiplication and Fast Fourier Transform. Graph representation. Exploring graphs and digraphs by BFS and (detailed) DFS. Acyclicity test (with proof), topological ordering. Strongly connected components by Kosaraju's algorithm (with proof). Shortest paths by Dijkstra's algorithm (with proof), priority queues as auxiliary data structure. The greedy paradigm. Minimum spanning trees by Kruskal's algorithm (with union-find data structure) and the Prim/Jarnik algorithm (with correctness proof). Huffman encoding, with priority queue, correctness proof. The dynamic programming paradigm. Examples: edit distance, chain matrix multiplication, knapsack with and without repetition, shortest paths (Floyd-Warshall and Bellman-Ford). Polynomial search problems, class NP, NP-complete problems. Significance of the notion. Central examples: Satisfiability, Clique, vertex cover, traveling salesperson, graph coloring.

### Medienformen

Blackboard, slide projection, exercise sheets, Moodle platform for communication.

### Literatur

- S. Dasgupta, C. H. Papadimitriou, U. V. Vazirani, Algorithms, McGraw Hill, 2006 (Prime textbook)
- T. Cormen, C. Leiserson, R. Rivest, C. Stein, Introduction to Algorithms, Second Edition, MIT Press 2001
- Sedgewick, Algorithms, Addison Wesley (Any edition will do, with or without specific programming language.)

Detailangaben zum Abschluss

verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2016

## Control Engineering

Fachabschluss: Prüfungsleistung mündlich 30 min Art der Notengebung: Gestufte Noten  
Sprache: Englisch Pflichtkennz.: Pflichtfach Turnus: Wintersemester

Fachnummer: 100093 Prüfungsnummer: 2200319

Fachverantwortlich: Prof. Dr. Pu Li

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2212																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
	2	1	0																											

### Lernergebnisse / Kompetenzen

Die Studierenden können

- die Grundlagen, Problemstellungen und Methoden der Regelungs- und Systemtechnik/technische Kybernetik klassifizieren,
- Systembeschreibungen ableiten,
- Methoden zur Systemanalyse anwenden,
- die Stabilität sowie einschleifige Regelkreise für industrielle Prozesse analysieren.

The students are able to

- classify the fundamentals, problem definitions, and methods of control and systems engineering/technical cybernetics,
- derive system descriptions,
- apply methods of systems analysis,
- analyse the stability and single control loops for industrial processes.

### Vorkenntnisse

Grundlagen der Mathematik, Physik, Elektrotechnik, Mechanik  
Fundamentals of mathematics, physics, electrical engineering, and mechanics

### Inhalt

Modeling of linear processes:

- Modeling with differential equations
- Linearization of nonlinear systems
- State space model

Laplace transformation:

- Laplace transformation of typical functions
- Properties of Laplace transformation
- Transfer function

Analysis of control systems in time domain:

- Dynamics of different plants
- Responses due to typical input signals
- Functions of typical controller

Stability analysis

### Medienformen

Präsentation, Vorlesungsskript, Tafelanschrieb  
Presentation, lecture script, writing on the black board

### Literatur

R. C. Dorf, R. H. Bishop. Modern Control Systems. Pearson. 2005  
K. Ogata. Modern control engineering. Pearson. 2010

### Detailangaben zum Abschluss

Schriftliche Prüfung, 90 min.

Written exam, 90 min.

verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016



## Software & Systems Engineering

Fachabschluss: Prüfungsleistung alternativ

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 100091

Prüfungsnummer: 2200317

Fachverantwortlich: Prof. Dr. Armin Zimmermann

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2236																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
	2	1	0																											

### Lernergebnisse / Kompetenzen

**Fachkompetenz:** Die Studierenden verstehen fortgeschrittene Probleme und Lösungsansätze für den Entwurf komplexer Softwaresysteme und ihrer technischen Anwendungen. Sie kennen gängige funktionale und nichtfunktionale Anforderungen an Softwaresysteme und sind in der Lage, aus ihnen Entwurfsentscheidungen abzuleiten. **Methodenkompetenz:** Die Studierenden sind fähig, geeignete Entwurfs-, Modellierungs- und Bewertungsverfahren für komplexe Softwaresysteme auszuwählen und auf konkrete Problemstellungen anzuwenden. **Sozialkompetenz:** Die Studierenden sind in der Lage, selbständig größere thematische zusammenhängende Literaturabschnitte zu den Themen der Veranstaltung durchzuarbeiten und zu erschließen. Sie können aus der Literatur Vorträge zu ergänzenden Themen halten.

### Vorkenntnisse

Bachelor in Computer Science, Computer Engineering or equivalent

### Inhalt

Introduction to advanced topics in Software Engineering and Systems Engineering

- Introduction and Overview of Topics
- Systems Engineering
- Selected Topics in Software Engineering
- Selected Topics in Model-Based Systems Engineering (Performance Evaluation)

RCSE students have priority

### Medienformen

Presentation slides, literature

### Literatur

Sommerville: Software Engineering (9th revised edition. International Version)  
 M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli and G. Franceschinis: Modelling with Generalized Stochastic Petri Nets (Wiley 1995)  
 INCOSE Systems Engineering Handbook, 2000  
 Blanchard, Fabrycky: Systems Engineering and Analysis (Prentice Hall 2006)  
 Cassandras/Lafortune: Introduction to Discrete Event Systems

### Detailangaben zum Abschluss

Successful completion and grading is based on

- 80% oral exam (25 min)
- 20 % work in the seminar

verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2012  
 Master Research in Computer & Systems Engineering 2016



## Advanced Mobile Communication Networks

Fachabschluss: Prüfungsleistung alternativ

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 100500

Prüfungsnummer: 2200348

Fachverantwortlich: Prof. Dr. Andreas Mitschele-Thiel

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2235																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
				2	1	0																								

### Lernergebnisse / Kompetenzen

The course introduces students in advanced topics in mobile data communication. It enables students to understand the research issues from a protocol- and system point of view, resulting from the mobility and the wireless transmission.

### Vorkenntnisse

Bachelor degree, basics of communication networks

### Inhalt

- Introduction
- Medium Access Schemes
- Mobility Management
- TCP/IP
- Self-Organization
- IEEE 802.11
- Quality of Service
- Ad Hoc Networks
- Cognitive Radio Networks
- Overview on cellular systems

### Medienformen

Presentations

### Literatur

We will provide it in class/class material.

### Detailangaben zum Abschluss

- The course consists of two parts: In the first part of the semester, lectures on the material are given. In the second part, individual studies (semester-long research projects that include a term paper and a presentation) help to improve understanding of the material.
- Grading scheme: 40% term paper plus presentation, 60% oral exam (20 min, registration at ICS office in Z1031).
- Binding registration for the exam (using Thoska or the registration form provided by the examination office IA) is required at the beginning of each semester (check the registration time window which is defined each semester) in order to participate in individual studies projects and the oral exam. As your course grade is a result of the individual studies and the oral exam, only formally registered students are eligible for participation in the individual studies and may receive credits for it.
  - If a student cannot pass the individual studies, he/she cannot attend for the oral exam.
  - To pass the individual studies, students must receive a permission from their supervisors to attend in the seminar. Other than, students will fail the course.
  - The second attempt of oral exam in each semester is just for the students who failed in the first attempt (not for any grade improvement or the students who were sick for the first attempt).

verwendet in folgenden Studiengängen:

Master Communications and Signal Processing 2013

Master Informatik 2013

Master Ingenieurinformatik 2009

Master Ingenieurinformatik 2014

Master Research in Computer & Systems Engineering 2016

## Cellular Communication Systems

Fachabschluss: Prüfungsleistung alternativ

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 100501

Prüfungsnummer: 2200349

Fachverantwortlich: Prof. Dr. Andreas Mitschele-Thiel

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2235																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							2 1 0																							

### Lernergebnisse / Kompetenzen

The course introduces students into the functionalities of cellular communication systems, esp. GSM/GPRS/EDGE, UMTS/HSPA, LTE/SAE. It enables students to understand network and protocol aspects of these system as well as aspects related to their deployment and management. Main topics are the network architecture, network elements, protocols, and services of these systems. The course allows students to understand main functions as mobility management, radio resource allocation, session management and QoS, as well as authentication, authorisation and network management.

### Vorkenntnisse

Communication protocols and networks, basics of mobile communication networks

### Inhalt

- Review of mobile communication basics
- Overview on GSM and GPRS
- UMTS architecture (mobility management, connection and session management, wideband CDMA, management of radio resources)
- UMTS radio access network
- High-Speed Packet Access (HSPA)
- Long-Term Evolution (LTE)
- System Architecture Evolution (SAE)
- Self-organization in LTE

### Medienformen

Presentations with beamer, presentation slides

### Literatur

- Kaaranen, Ahtiainen, Laitinen, Naghian, Niemi. UMTS Networks – Architecture, Mobility and Services. Wiley, 2001
- Holma, Toskala. WCDMA for UMTS. revised edition, Wiley, 2002
- Dahlmann, Parkvall, Sköld. 4G: LTE/LTE-Advanced for Mobile Broadband, AP, 2011
- Stefania Sesia, Issam Toufik, Matthew Baker. LTE - The UMTS Long Term Evolution: From Theory to Practice

### Detaillangaben zum Abschluss

- The course consists of two parts: In the first part of the semester, lectures on the material are given. In the second part, individual studies (semester-long research projects that includes a presentation) help to improve understanding of the material.
- Grading scheme: 20% individuell studies, 80% oral exam (20 min, registration at ICS office in Z1031).
- Binding registration for the exam (using Thoska or the registration form provided by the examination office IA) is required at the beginning of each semester (check the registration time window which is defined each semester) in order to participate in individual studies projects and the oral exam. As your course grade is a result of the individual studies and the oral exam, only formally registered students are eligible for participation in the individual studies and may receive credits for it.
- The second attempt of oral exam in each semester is just for the students who failed in the first attempt (not

for any grade improvement or the students who were sick for the first attempt).

verwendet in folgenden Studiengängen:

Master Communications and Signal Processing 2013

Master Informatik 2013

Master Ingenieurinformatik 2009

Master Ingenieurinformatik 2014

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016



verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016

## Distributed Data Management

Fachabschluss: Prüfungsleistung mündlich 30 min

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 101155

Prüfungsnummer: 2200457

Fachverantwortlich: Prof. Dr. Kai-Uwe Sattler

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2254																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
				2	1	0																								

### Lernergebnisse / Kompetenzen

Nachdem Studierende diese Veranstaltung besucht haben, kennen sie die Grundlagen verteilter und paralleler Datenmanagementlösungen. Sie verstehen die Prinzipien dieser Techniken und können darauf aufbauend selbst Lösungen entwickeln. Die Studierenden können Techniken zur Anfrageverarbeitung, Replikation und Konsistenzsicherung erklären und hinsichtlich ihrer Vor- und Nachteile für verschiedene Einsatzzwecke bewerten.

Sie sind in der Lage, verteilte Datenbanken zu entwerfen und aktuelle Datenbanktechnologien verteilter und paralleler Systeme zu bewerten und anzuwenden

### Vorkenntnisse

Vorlesung Datenbanksysteme, Transaktionale Informationssysteme

### Inhalt

Einführung und Motivation; Grundlagen verteilter Datenbanken: Architektur und Datenverteilung, verteilte Anfrageverarbeitung, Replikationsverfahren; Parallele Datenbanksysteme: Architektur und Datenverteilung, parallele Anfrageverarbeitung, Shared-Disk-Systeme; Web-Scale Data Management: SaaS und Multi Tenancy, Virtualisierungstechniken, Konsistenzmodelle, QoS, Partitionierung, Replikation, DHTs, MapReduce

### Medienformen

Vorlesung mit Präsentationen und Tafel, Handouts, Moodle

### Literatur

E. Rahm: Mehrrechner-Datenbanksysteme, Addison-Wesley, Bonn, 1994

M. Tamer Özsu, P. Valduriez: Principles of Distributed Database Systems, 3. Auflage, Springer, 2011

C. T. Yu, W. Meng: Principles of Database Query Processing for Advanced Applications, Morgan Kaufmann Publishers, San Francisco, Ca, 1998

Lehner, Sattler: Web-Scale Data Management for the Cloud, Springer, 2013

### Detaillangaben zum Abschluss

mündliche Prüfung (30 min)

### verwendet in folgenden Studiengängen:

Master Informatik 2013

Master Ingenieurinformatik 2009

Master Ingenieurinformatik 2014

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016

Master Wirtschaftsinformatik 2013

Master Wirtschaftsinformatik 2014

Master Wirtschaftsinformatik 2015

Master Wirtschaftsinformatik 2018

## Software Architecture

Fachabschluss: Prüfungsleistung alternativ

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 101870

Prüfungsnummer: 2200615

Fachverantwortlich: Dr. Detlef Streitferdt

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):128			SWS:2.0																					
Fakultät für Informatik und Automatisierung						Fachgebiet:223																								
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							2 0 0																							

### Lernergebnisse / Kompetenzen

Students will learn where to use software engineering tools within a given project context. They learn to assess the quality of an architecture. They understand software architecture patterns.

Students know about development processes and ~tools. They learn to estimate development efforts.

Students learn how to make use of development processes in groups/teams. They learn about soft factors and their impact on software development projects.

### Vorkenntnisse

Object-orientation, UML, OO-Programming

C++ and/or Java

### Inhalt

This lecture presents software engineering methods and tools. Development activities are embedded in development processes. The lecture teaches students about software architecture goals, ~patterns, the quality of architectures and how to assess this quality.

### Medienformen

Slides, PDF-documents, HTML-pages

### Literatur

#### General

[Fowl 1999] Martin Fowler, "Refactoring – Improving the Design of Existing Code", Addison Wesley, 1999.

[Gamm 1995] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns – Elements of Reusable Object-Oriented Software", Addison Wesley, 1995.

[Mart 2009] Robert C. Martin, „Clean Code“, Prentice Hall, 2009.

[McCo 2004] Steve McConnell, „Code Complete 2nd Edition“, Microsoft Press, 2004.

[Somm 2007] Ian Sommerville, "Software Engineering", Pearson Studium, 2007.

[Mens 2008] T. Mens and S. Demeyer, Eds., "Software Evolution". Springer-Verlag New York Inc, 2008.

#### Special Topics ...

#### Development processes

[Beck 2000] Kent Beck, „eXtreme Programming eXplained“, Addison Wesley, 2000.

[Carr 1993] Marvin J. Carr, Suresh L. Konda, Ira Monarch, F. Carol Ulrich, Clay F. Walker, "Taxonomy-Based Risk Identification", Carnegie Mellon University, Technical Report CMU/SEI-93-TR-6, ESC-TR-93-183, 1993.

[Open 2011] Eclipse Process Framework, "Open Unified Process, OpenUP", content retrieved 2011-10-01, 2011.

#### Requirements

[Bere 2009] Brian Berenbach, Daniel J. Paulish, Juergen Kazmeier, Arnold Rudorfer, "Software &amp; Systems Requirements Engineering In Practice", Mc Graw Hill, 2009.

[Haya 1990] S. I. Hayakawa, "Language in Thought and Action", Harvest Books, 1990.

[KoSo 1998] Gerald Kotonya, Ian Sommerville, "Requirements Engineering - Processes and Techniques", John Wiley &amp; Sons, 1998.

[Kula 2000] Daryl Kulak, Eamonn Guiney, "Use Cases - Requirements in Context", Addison-Wesley, 2000.

[Lams 2001] Axel van Lamsweerde, "Goal-Oriented Requirements Engineering: A Guided Tour", in Proceedings of the 5th IEEE International Symposium on Requirements Engineering (RE 2001), 27-31 August 2001, Toronto, Canada, 2001.

[Lams 2009] Axel van Lamsweerde, "Requirements Engineering: From System Goals to UML Models to



Software Specifications", John Wiley & Sons, 2009.

[McCo 2006] Steve McConnell, "Software Estimation", Microsoft Press, 2006.

[Robe 1999] Suzanne Robertson, James Robertson, "Mastering the Requirements Process", Addison-Wesley, 1999.

[Schu 2000] G. Gordon Schulmeyer, Garth R. Mackenzie, "Verification & Validation of Modern Software-Intensive Systems", Prentice Hall, 2000.

[SoSa 1997] Ian Sommerville, Pete Sawyer, "Requirements Engineering: A Good Practice Guide", John Wiley & Sons, 1997.

[Wieg 1999] Karl E. Wiegers, "Software Requirements", Microsoft Press, 1999.

[With 2007] Stephen Withall, "Software Requirement Patterns", Microsoft Press, 2007.

Architecture, Product Lines

[Brow 2011] Amy Brown, Greg Wilson (ed.) "The Architecture of Open Source Applications", Vol. 1, <http://aosabook.org>, 2011.

[Brow 2012] Amy Brown, Greg Wilson (ed.) "The Architecture of Open Source Applications", Vol. 2, <http://aosabook.org>, 2012.

[Clem 2002] Paul Clements, Rick Kazman, Mark Klein, "Evaluating Software Architectures", Addison Wesley, 2002.

[Kang 1990] K. Kang, S. Cohen, J. Hess, W. Novak, and A. Peterson, "Feature-Oriented Domain Analysis (FODA) Feasibility Study", SEI Institute, Carnegie Mellon University, USA, CMU/SEI-90-TR-021, 1990.

[Kazm 2000] Rick Kazman, Mark Klein, Paul Clements, "ATAM: Method for Architecture Evaluation", TECHNICAL REPORT, CMU/SEI-2000-TR-004, ESC-TR-2000-004, 2000.

[Lind 2007] F. J. van der Linden, K. Schmid, and E. Rommes, "Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering". Berlin: Springer, 2007.

[Love 2005] Robert Love, "Linux Kernel Development (2nd Edition)", Novell Press, 2005.

[Pohl 2005] Klaus Pohl, Günter Böckle, Frank van der Linden, "Software Product Line Engineering – Foundations, Principles, and Techniques", Springer, Heidelberg 2005.

[Spin 2009] D. Spinellis and G. Gousios, "Beautiful Architecture: Leading Thinkers Reveal the Hidden Beauty in Software Design". O'Reilly Media, 2009.

Computer Science add-ons

[Bern 2003] William Bernbach, "A Technique for Producing Ideas", McGraw-Hill, 2003.

[Broo 1995] Frederick P. Brooks, Jr., "The Mythical Man Month", Addison-Wesley, 1995.

[Mich 2006] Michael Michalko, "Thinkertoys: A Handbook of Creative-Thinking Techniques", Ten Speed Press, 2006.

[Your 1997] Edward Yourdon, "Death March", Prentice-Hall, 1997.

#### Detailangaben zum Abschluss

Students need to analyse an Open Source project and prepare an architecture document and a presentation of this architecture. This work (document+presentation) accounts for 50% of the final mark. Both parts, the document+presentation and the final, oral exam are required to finish this lecture.

verwendet in folgenden Studiengängen:

Master Medientechnologie 2017

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016



## Systems Optimization

Fachabschluss: Prüfungsleistung mündlich 30 min Art der Notengebung: Gestufte Noten  
Sprache: Englisch Pflichtkennz.: Pflichtfach Turnus: Wintersemester

Fachnummer: 100965

Prüfungsnummer: 2200416

Fachverantwortlich: Prof. Dr. Pu Li

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):105			SWS:4.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2212																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							2 2 0																							

### Lernergebnisse / Kompetenzen

• to model and classify optimization problems  
• to identify relevant optimization algorithms and solve real-life engineering optimization problems  
• to enable the student solve practical optimization problems using modern software tools  
• to enable the student analyze the viability of optimization solutions for practical use

### Vorkenntnisse

Fundamentals of Mathematics and Control Engineering

### Inhalt

#### PRELIMINARIES

1. Introduction, Motivation, and Preliminaries
  - Importance of Systems Optimization
  - Mathematical Preliminaries
  - Convex sets and Convex Functions

#### PART - I : Steady-State Optimization Problems and Applications

##### Methods of Unconstrained Optimization Problems

- 2.1. First- and Second-Order Optimality Conditions
- 2.2. The Method of Steepest Descent
- 2.3. The Newton Method
- 2.4. The Levenberg-Marquardt Method
- 2.5. Quasi-Newton Methods
- 2.6. Line-search Methods
- 2.7. System of Nonlinear Equations
  - 2.7a. Numerical Algorithms for Systems of Nonlinear Equations
  - 2.7b. Numerical Solution Methods for Differential Algebraic Equations: backward differentiation formula (BDF), Single and Multiple-shooting, collocation methods

##### Methods of Constrained Optimization Problems

- 3.1. The Karush-Kuhn-Tucker Optimality Conditions
- 3.2. Convex Optimization Problems
- 3.3. Penalty Methods
- 3.4. Barrier and Interior-Point Methods
- 3.5. The Sequential Quadratic Programming (SQP) Method

#### Part-II: Dynamic Optimization Problems and Applications

4. Introduction to Dynamic Optimization
5. Direct Methods for Dynamic Optimization Problems
  - 5.1. Collocation Methods for Dynamics Optimization Problems
  - 5.2. Costate estimation
6. Introduction to Model-Predictive Control

#### Appendix

- A review on numerical linear algebra methods
- Introduction to 1D quadrature rules and orthogonal polynomial collocation
- A review on numerical methods of Ordinary Differential Equations (ODEs): numerical methods of initial value and boundary value ordinary differential equations - Euler method, Runge-Kutta, BDF, implicit Runge-Kutta,
- A brief introduction differential Algebraic Equations (DAEs) and Applications: The concept of Index in DAEs, Consistent Initialization, etc.
- A summary of the Classical Theory of Optimal Control Problems - the Pontryagin Principle - Indirect Methods

#### Medienformen

Presentation, Lecture slides script, blackboard presentation

#### Literatur

• J.T. Betts: Practical methods for optimal control using nonlinear programming, SIAM 2001.  
 • A. E. Bryson, Y.-C. Ho: Applied optimal control : optimization, estimation, and control, Taylor & Francis, 1975.  
 • C. Chiang: Elements of dynamic optimization. McGraw-Hill, 1992.  
 • E. Eich-Soellner, C. Führer: Numerical methods in multibody dynamics. B.G Teubner, 1998.  
 • M. Gerdt: Optimal control of ODEs and DAEs. De Gruyter, 2012.  
 • D.R. Kirk: Optimal Control theory: an introduction. Dover Publisher, 2004.  
 • J. Nocedal, S.J. Wright: Numerical methods of optimization. 2nd ed. Springer Verlag 2006.  
 • R.D. Rabinet III et al.: Applied dynamic programming for optimization of dynamical systems. SIAM 2005.  
 • S.S. Rao: Engineering optimization - theory and practice. Wiley, 1996.

#### Detailangaben zum Abschluss

Oral examination, 30 min.

#### verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2012  
 Master Research in Computer & Systems Engineering 2016

## Systems Security

Fachabschluss: Prüfungsleistung mündlich 20 min Art der Notengebung: Gestufte Noten  
Sprache: Englisch Pflichtkennz.: Pflichtfach Turnus: Sommersemester

Fachnummer: 100963 Prüfungsnummer: 2200414

Fachverantwortlich: Prof. Dr. Winfried Kühnhauser

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2255																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
				2	1	0																								

### Lernergebnisse / Kompetenzen

This course focuses on the basic paradigms, methods and concepts in the field of model-based security engineering - the methodical process of engineering a computer system's security properties based on formal security models. In an early stage of the engineering process formal security models are used for the precise and unambiguous representation of security policies which then are analyzed by static model checking and simulative model execution. Successful models afterwards are transformed via specification languages into executable code which finally is integrated into a system's trusted computing base.

Students acquire knowledge on methodological engineering of security properties based on security policies and their formal models. Successful students will be able to design, analyze and specify security policies and integrate them into the trusted computing base of IT systems.

### Vorkenntnisse

BSc Computer Science, especially algorithms and complexity, automata and formal languages, operating systems, networks, discrete structures

### Inhalt

Course topics are

- Security Requirements analysis
- Security policies and formal security models
- Model engineering and analysis
- Policy specification languages
- Security mechanisms
- Trusted computing bases, security architectures and reference monitors

### Medienformen

Presentations with projector and whiteboard, books, papers, assignments, discussions

### Literatur

- William Stallings, Lawrie Brown: Computer Security. Pearson, 2nd Edition, 2012, 810 pages.
- Matthew Bishop: Computer Security: Art and Science. Addison-Wesley Educational Publishers Inc, 2012 (2. Edition), 1168 pages.
- Trent Jaeger: Operating System Security. Synthesis Lectures on Information Security, Privacy and Trust #1, Morgan & Claypool Publishers, 2008.
- N. Akosan et. al.: Mobile Platform Security. Synthesis Lectures on Information Security, Privacy and Trust #9, Morgan & Claypool Publishers, 2014.
- Anupam Datta et. al.: Analysis Techniques for Information Security. Synthesis Lectures on Information Security, Privacy and Trust #2, Morgan & Claypool Publishers, 2010.
- Ross Anderson: Security Engineering. John Wiley & Sons, 2nd Edition, 2008, 1040 pages. Also available online.
- Frank Mayer, Karl Macmillan, David Caplan: SELinux by Example. Prentice Hall 2007, 425 pages.
- Bruce Schneier: Secrets and Lies - Digital Security in a Networked World. John Wiley & Sons 2000, 408 pages.

### Detailangaben zum Abschluss

direct grade

verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016

## Lab Training

Fachabschluss: Studienleistung

Art der Notengebung: Testat / Generierte

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: ganzjährig

Fachnummer: 100094

Prüfungsnummer: 2200320

Fachverantwortlich: Dr. Siegbert Hopfgarten

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):116			SWS:3.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2212																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
				0	0	3																								

### Lernergebnisse / Kompetenzen

Unter Einbeziehung der in den Basic Studies gelehrt, der im laufenden Fachsemester angebotenen Fächer und Module sowie im Selbststudium erworbener Kenntnisse wenden die Studierenden fachspezifische Kenntnisse an. Individuell oder in kleinen Gruppen werden die Problemstellungen analysiert, Lösungswege generiert und an praxisnahen Versuchsaufbauten und Aufgabenstellungen getestet. Sie beurteilen die erzielten Ergebnisse.

Including both the subjects and modules, taught in Basic Studies, in the running term/semester and knowledge acquired in self-studies, the students apply subject-specific knowledge. Problems are analyzed, solutions are generated and tested at lab experiments with practical orientation individually or in small groups. The results achieved are evaluated.

### Vorkenntnisse

Basic Studies, Advanced Studies

### Inhalt

- 1) Embedded Systems/Programming
  - 2) Database lab
  - 3) Identification and control
- (zwei von drei Einheiten müssen gewählt werden)  
(two out of three units have to be chosen)  
(Beteiligung von insgesamt drei Fachgebieten der Fakultät IA)  
(Participation of three groups of the faculty IA)

### Medienformen

Versuchsanleitungen, Aufgabenstellungen zur Lösung an PCs/eingebetteten Systemen/mittels Datenbanken/Labora aufbauten  
Lab experiments instructions, formulated tasks to be solved at PCs/embedded systems/data bases/lab experiments

### Literatur

F. M. Mims. Started in Electronics. Master Publishing Inc. 2003.  
J. Sanchez, M. P. Canton. Embedded Systems Circuits and Programming. CRC Press. 2012  
R. C. Dorf, R. H. Bishop. Modern Control Systems. Pearson. 2005

### Detailangaben zum Abschluss

Unbenoteter Schein (Testat) für zwei von drei der angebotenen Einheiten (siehe Inhalt).  
Not marked certificate for two out of three units (see content).

verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2012  
Master Research in Computer & Systems Engineering 2016

## Research Seminar RCSE

Fachabschluss: Prüfungsleistung alternativ

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 8018

Prüfungsnummer: 2200605

Fachverantwortlich: Prof. Dr. Kai-Uwe Sattler

Leistungspunkte: 5			Workload (h):150			Anteil Selbststudium (h):128			SWS:2.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2254																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
				0	2	0																								

### Lernergebnisse / Kompetenzen

A research seminar serves to deepen the knowledge in dealing with scientific texts in a receptive and descriptive way. The main task of a student is the independent development of one or more pieces of scientific literature up to one's own understanding and the closed presentation of this material in a lecture to other students and the organizer, with questioning and discussion on a scientific level. A written summary shall be demanded. The seminar also serves to train the communication skills of the students in a subject-specific context.

### Vorkenntnisse

### Inhalt

wird zu Beginn der Veranstaltung bzw. bei der Ausschreibung bekannt gegeben.

### Medienformen

Vortrag, bereitgestellte Literatur

### Literatur

wird zu Beginn der Veranstaltung bzw. bei der Ausschreibung bekannt gegeben

### Detailangaben zum Abschluss

### verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2009  
Master Research in Computer & Systems Engineering 2012  
Master Research in Computer & Systems Engineering 2016





## Research Project

Fachabschluss: Prüfungsleistung alternativ

Art der Notengebung: Gestufte Noten

Sprache: Englisch

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 8016

Prüfungsnummer: 2200288

Fachverantwortlich: Prof. Dr. Armin Zimmermann

Leistungspunkte: 15			Workload (h):450			Anteil Selbststudium (h):405			SWS:4.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2236																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							0 4 0																							

### Lernergebnisse / Kompetenzen

Die Studierenden sind in der Lage, unter Anleitung selbständig aktuelle Forschungsthemen zu bearbeiten. Sie können offene Probleme analysieren, den Stand der Technik erarbeiten und Vorschläge für neuartige Lösungen entwickeln und realisieren. Die Studierenden sind in der Lage, zu aktuellen Forschungsfragen beizutragen und ihre Ergebnisse zu präsentieren sowie einzuordnen.

### Vorkenntnisse

Basic studies of RCSE curriculum and research skills seminar

### Inhalt

Forschungsarbeiten innerhalb der beteiligten Fachgebiete Research work within the participating groups

### Medienformen

abhängig vom individuellen Projekt depends on the actual project

### Literatur

abhängig vom individuellen Projekt depends on the actual project

### Detailangaben zum Abschluss

A report document and a presentation talk about the results. Depending on the topic there may be additional requirements (for example software implementation, experimental results, demonstration ..).

### verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2009

Master Research in Computer & Systems Engineering 2012

Master Research in Computer & Systems Engineering 2016

## Internship

Fachabschluss: Studienleistung alternativ Art der Notengebung: Testat / Generierte  
Sprache: English Pflichtkennz.: Pflichtfach Turnus: ganzjährig

Fachnummer: 101722

Prüfungsnummer: 2200607

Fachverantwortlich: Prof. Dr. Kai-Uwe Sattler

Leistungspunkte: 15			Workload (h):450			Anteil Selbststudium (h):405			SWS:4.0																					
Fakultät für Informatik und Automatisierung								Fachgebiet:2254																						
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							min. 10																							

### Lernergebnisse / Kompetenzen

applicability of own skills and motivations to a particular field of application in the IT industry by solving specific tasks; overview of general aspects of industrial environment, such as team work, working to a deadline, economic efficiency, quality management, data protection; continued development of soft skills

### Vorkenntnisse

/

### Inhalt

Practical activities in which scientific methods are put into practice within an industrial or practical context of IT companies or organizations with the aim to conceptualize, implement, evaluate, utilize, and maintain complex computer systems

### Medienformen

/

### Literatur

/

### Detailangaben zum Abschluss

verwendet in folgenden Studiengängen:

Master Research in Computer & Systems Engineering 2016

## Modul: Soft Skills RCSE

Modulnummer: 8024

Modulverantwortlich: Prof. Dr. Winfried Kühnhauser

Modulabschluss:

### Lernergebnisse

This module trains skills that help to do efficiently perform the characteristic patterns of scientific work. The students learn and train to present scientific results in reading and writing, to design conference posters, to write research proposals, to review papers, and to organize workshops and conferences. Additionally, skills in German language are trained.

### Vorraussetzungen für die Teilnahme

siehe individuelle Fächerbeschreibungen

### Detailangaben zum Abschluss



## Modul: Masterarbeit RCSE

Modulnummer: 8027

Modulverantwortlich: Prof. Dr. Kai-Uwe Sattler

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

Die Studierenden vertiefen in einem speziellen fachlichen Thema ihre bisher erworbenen Kompetenzen. Sie werden befähigt, eine komplexe und konkrete Problemstellung zu beurteilen und unter Anwendung der bisher erworbenen Theorie- und Methodenkompetenzen selbständig zu bearbeiten. Das Thema ist gemäß wissenschaftlicher Standards zu dokumentieren und die Studierenden werden befähigt, entsprechende wissenschaftlich fundierte Texte zu verfassen. Die Studierenden erwerben Problemlösungskompetenz und lernen es, die eigene Arbeit zu bewerten und einzuordnen.

### Voraussetzungen für die Teilnahme

Zulassung zur Masterarbeit durch den Prüfungsausschuss

### Detailangaben zum Abschluss

## Kolloquium zur Master-Arbeit

Fachabschluss: Prüfungsleistung mündlich 30 min

Art der Notengebung: Gestufte Noten

Sprache: Deutsch und Englisch

Pflichtkennz.: Pflichtfach

Turnus: ganzjährig

Fachnummer: 101482

Prüfungsnummer: 99002

Fachverantwortlich: Prof. Dr. Kai-Uwe Sattler

Leistungspunkte: 6			Workload (h):180			Anteil Selbststudium (h):180			SWS:0.0																					
Fakultät für Informatik und Automatisierung									Fachgebiet:2254																					
SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										180 h																				

### Lernergebnisse / Kompetenzen

Die Studierenden vertiefen in einem speziellen fachlichen Thema ihre bisher erworbenen Kompetenzen. Sie werden befähigt, eine komplexe und konkrete Problemstellung zu beurteilen und unter Anwendung der bisher erworbenen Theorie- und Methodenkompetenzen selbstständig zu bearbeiten. Das Thema ist gemäß wissenschaftlicher Standards zu dokumentieren und die Studierenden werden befähigt, entsprechende wissenschaftlich fundierte Texte zu verfassen. Die Studierenden erwerben Problemlösungskompetenz und lernen es, die eigene Arbeit zu bewerten und einzuordnen.

### Vorkenntnisse

Zulassung zur Masterarbeit durch den Prüfungsausschuss

### Inhalt

siehe Modulbeschreibung

### Medienformen

wissenschaftlicher Vortrag

### Literatur

Literatur wird mit Ausgabe des Themas bekannt gegeben oder ist selbstständig zu recherchieren.

### Detaillangaben zum Abschluss

### verwendet in folgenden Studiengängen:

Master Research in Computer &amp; Systems Engineering 2009

Master Research in Computer &amp; Systems Engineering 2012

Master Research in Computer &amp; Systems Engineering 2016

## MaA wissenschaftliche Arbeit

Fachabschluss: Masterarbeit schriftlich 6 Monate

Art der Notengebung: Generierte Noten

Sprache: Deutsch und Englisch

Pflichtkennz.: Pflichtfach

Turnus: ganzjährig

Fachnummer: 101478

Prüfungsnummer: 99001

Fachverantwortlich: Prof. Dr. Kai-Uwe Sattler

Leistungspunkte: 24

Workload (h): 720

Anteil Selbststudium (h): 720

SWS: 0.0

Fakultät für Informatik und Automatisierung

Fachgebiet: 2254

SWS nach Fach- semester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS			8.FS			9.FS			10.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P

720 h

### Lernergebnisse / Kompetenzen

Die Studierenden vertiefen in einem speziellen fachlichen Thema ihre bisher erworbenen Kompetenzen. Sie werden befähigt, eine komplexe und konkrete Problemstellung zu beurteilen und unter Anwendung der bisher erworbenen Theorie- und Methodenkompetenzen selbstständig zu bearbeiten. Das Thema ist gemäß wissenschaftlicher Standards zu dokumentieren und die Studierenden werden befähigt, entsprechende wissenschaftlich fundierte Texte zu verfassen. Die Studierenden erwerben Problemlösungskompetenz und lernen es, die eigene Arbeit zu bewerten und einzuordnen.

### Vorkenntnisse

Zulassung zur Masterarbeit durch den Prüfungsausschuss

### Inhalt

siehe Modulbeschreibung

### Medienformen

wissenschaftlicher Vortrag

### Literatur

Literatur wird mit Ausgabe des Themas bekannt gegeben oder ist selbstständig zu recherchieren.

### Detailangaben zum Abschluss

### verwendet in folgenden Studiengängen:

Master Research in Computer &amp; Systems Engineering 2009

Master Research in Computer &amp; Systems Engineering 2012

Master Research in Computer &amp; Systems Engineering 2016





## **Glossar und Abkürzungsverzeichnis:**

LP	Leistungspunkte
SWS	Semesterwochenstunden
FS	Fachsemester
V S P	Angabe verteilt auf Vorlesungen, Seminare, Praktika
N.N.	Nomen nominandum, Platzhalter für eine noch unbekannte Person (wikipedia)
Objekttypen lt. Inhaltsverzeichnis	K=Kompetenzfeld; M=Modul; P,L,U= Fach (Prüfung, Lehrveranstaltung, Unit)